**Front matter**

lang: ru-RU  
title: "Лабораторная работа №7"  
subtitle: "Дисциплина: операционные системы"  
author: "Бирюкова Анастасия Анатольевна"

**Formatting**

toc-title: "Содержание"  
toc: true # Table of contents  
toc\_depth: 2  
lof: true # List of figures  
lot: true # List of tables  
fontsize: 12pt  
linestretch: 1.5  
papersize: a4paper  
documentclass: scrreprt  
polyglossia-lang: russian  
polyglossia-otherlangs: english  
mainfont: PT Serif  
romanfont: PT Serif  
sansfont: PT Sans  
monofont: PT Mono  
mainfontoptions: Ligatures=TeX  
romanfontoptions: Ligatures=TeX  
sansfontoptions: Ligatures=TeX,Scale=MatchLowercase  
monofontoptions: Scale=MatchLowercase  
indent: true  
pdf-engine: lualatex  
header-includes:  
- \linepenalty=10 # the penalty added to the badness of each line within a paragraph (no associated penalty node) Increasing the value makes tex try to have fewer lines in the paragraph.  
- \interlinepenalty=0 # value of the penalty (node) added after each line of a paragraph.  
- \hyphenpenalty=50 # the penalty for line breaking at an automatically inserted hyphen  
- \exhyphenpenalty=50 # the penalty for line breaking at an explicit hyphen  
- \binoppenalty=700 # the penalty for breaking a line at a binary operator  
- \relpenalty=500 # the penalty for breaking a line at a relation  
- \clubpenalty=150 # extra penalty for breaking after first line of a paragraph  
- \widowpenalty=150 # extra penalty for breaking before last line of a paragraph  
- \displaywidowpenalty=50 # extra penalty for breaking before last line before a display math  
- \brokenpenalty=100 # extra penalty for page breaking after a hyphenated line  
- \predisplaypenalty=10000 # penalty for breaking before a display  
- \postdisplaypenalty=0 # penalty for breaking after a display  
- \floatingpenalty = 20000 # penalty for splitting an insertion (can only be split footnote in standard LaTeX)  
- \raggedbottom # or \flushbottom  
- \usepackage{float} # keep figures where there are in the text

**- \floatplacement{figure}{H} # keep figures where there are in the text**

**Цель работы**

Ознакомление с инструментами поиска файлов и фильтрации текстовых данных. Приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

**Задание**

1. Осуществите вход в систему, используя соответствующее имя пользователя.

2. Запишите в файл file.txt названия файлов, содержащихся в каталоге /etc.  
Допишите в этот же файл названия файлов, содержащихся в вашем домашнем  
каталоге.

3. Выведите имена всех файлов из file.txt, имеющих расширение .conf, после  
чего запишите их в новый текстовой файл conf.txt.

4. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа c? Предложите несколько вариантов, как это сделать.

5. Выведите на экран (постранично) имена файлов из каталога /etc, начинающиеся с символа h.

6. Запустите в фоновом режиме процесс, который будет записывать в файл  
~/logfile файлы, имена которых начинаются с log.

7. Удалите файл ~/logfile.

8. Запустите из консоли в фоновом режиме редактор gedit.

9. Определите идентификатор процесса gedit, используя команду ps, конвейер и  
фильтр grep. Можно ли определить этот идентификатор более простым способом?

10. Прочтите справку (man) команды kill, после чего используйте её для завершения процесса gedit.

11. Выполните команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man.

12. Воспользовавшись справкой команды find, выведите имена всех директорий,  
имеющихся в вашем домашнем каталоге.

**Выполнение лабораторной работы**

1. Вход в систему.

2. Запишем в файл file.txt названия файлов, содержащихся в каталоге /etc(используем команду ls /etc > file.txt). (Рис.1-3) Допишем в этот же файл названия файлов, содержащихся в нашем домашнем каталоге. Использовали программу ls >> file.txt  
Рис(4-5)

3. Выведем имена всех файлов из file.txt имеющих расширение .conf, использовав команду grep .conf file.txt  
Рис 6-7.  
  
  
Затем запишем их в новый текстовой файл conf.txt.(Рис.8)

4. Определим, какие файлы в нашем домашнем каталоге имеют имена, начинавшиеся с символа c.  
Первый вариант, используя команду find ~ -name "c\*" -print(Рис.9-10)

Второй способ ls -l | grep c\*(Рис.11)

1. Выведем на экран имена файлов из каталога /etc, начинающиеся с символа h. Используем команду find /etc -name 'h\*' -print | more(Рис.12-13)

2. Запустим в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log. Используем команду find ~ -name "log\*" -print > logfile &(Рис.14)

3. Удалим файл ~/logfile, используя команду rm -i ~/logfile(Рис.15)

4. Запустим в фоновом режиме редактор gedit, используя команду gedit &.

5. Определим идентификатор процесса gedit используя команду ps, конвейер и фильтр grep. Использовали команду ps axu | grep gedit(Рис.16)

6. Прочтём справку (man) команды kill(Рис.17)

после чего используем ее для завершения процесса gedit, набрав команду kill % 6031 cd(Рис.18)  
  
11. Предварительно получив более подробную информацию о командах df и du, с помощью команды man, выполним их(Рис 19-20)  
  
  
Получим информацию о команде df, набрав man df, после наберём команду df  
Получим информацию о du, набрав man du, после наберём команду du и df(Рис.21-22)  
  
  
12. Воспользовавшись справкой команды find выведем имена всех директорий, имеющихся в нашем домашнем каталоге. Для этого используем команду find -type d(Рис.23-25)

**Выводы**

Мы ознакомились с инструментами поиска файлов и фильтрации текстовых данных. А также приобрели практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

**Контрольные вопросы**

1. – stdin — стандартный поток ввода (клавиатура),

– stdout — стандартный поток вывода (консоль),

– stderr — стандартный поток вывод сообщений об ошибках на экран

2.Символ < используется для переназначения стандартного ввода команды.

Символ >> используется для присоединения данных в конец файла стандартного вывода команды(файл открывается в режиме добавления)

3.Конвейер - способ связи между двумя программами.Конвейер (pipe) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передается последующей. Синтаксис следующий: команда1 | команда 2

4.Процесс - это программа, которая выполняется в отдельном виртуальном адресном пространстве. Когда пользователь регистрируется в системе, автоматически создается процесс, в котором выполняется оболочка (shell), например, /bin/bash.

Компьютерная программа сама по себе — это только пассивная совокупность инструкций, в то время как процесс — это непосредственное выполнение этих инструкций.

1. Process ID(PID) - идентификатор порожденного процесса. Group ID (GID-идентификация группы пользователей.

2. Запущенные фоном программы называются задачами (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент задач. Для завершения задачи необходимо выполнить команду :

kill %номер задачи

1. top — консольная команда, которая выводит список работающих в системе процессов и информации о них. По умолчанию она в реальном времени сортирует их по нагрузке на процессор.

htop — компьютерная программа, предназначенная для вывода на терминал списка запущенных процессов и информации о них . Создана, как альтернатива программы top.

1. Команда find используется для поиска и отображения имен файлов,соответствующих заданной строке символов.

Формат команды:

find путь [-опции]

Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск.

Пример:

Вывести на экран имена файлов из вашего домашнего каталога и его подкаталогов, начинающихся на f:

find ~ -name "f\*" -print

где ~ — обозначение вашего домашнего каталога, -name — после этой опции указывается имя файла, который нужно найти, "f\*" — строка символов, определяющая имя файла, -print — опция, задающая вывод результатов поиска на экран.

9.Можно, команда grep способна обрабатывать стандартный вывод других команд (любой текст). Для этого следует использовать конвейер, связав вывод команды с вводом grep.

Пример:

Показать строки во всех файлах в вашем домашнем каталоге с именами начинающимися на f, в которых есть слово begin:

grep begin f\*

10.Команда df показывает размер каждого смонтированного раздела диска.

11.Команда du показывает число килобайт, используемое каждым файлом или каталогом.

1. Когда зависает процесс, следует нажать одновременно клавиши ctrl+alt+del, чтобы вызвать «Диспетчер задач», а после закрыть программу, которая не отвечает.